Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Previously Amended) A method of internally encrypting data in a relational database, comprising the steps of:

providing a database engine having encryption as a database kernel feature;

providing a security dictionary comprising one or more security catalogs;

receiving data from a user;

associating said data with a database column and at least one authorized user;

generating a working encryption key;

internally encrypting said working encryption key within said database engine using a public key from an authorized user;

storing said encrypted working key in a security catalog; and

internally encrypting said data within said database engine using said working key.

2. (Original) The method of claim 1 further comprising the step of generating a private key needed to decrypt said encrypted working key.

3. (Original) The method of claim 2 wherein said public key is a password and is used by the system to look up said private key.

4. (Original) The method of claim 1 wherein said step of associating said data with a database column and a user is accomplished with an extended SQL syntax and further comprises the step of creating a relational database object comprising:

the identity of said authorized users;

a relational database table;

the identity of said column within said relational database table; and

one or more security flags, said flags indicating user privileges to access said data.

5. (Original) The method of claim 1 wherein said working key is provided by a user.

6. (Original) The method of claim 1 wherein said working key is randomly generated.

7. (Original) The method of claim 1 further comprising the steps of:

receiving a query and private key from a user;

checking the ownership of an encrypted column using said security catalog to verify the user is authorized;

internally decrypting said encrypted working encryption key with said private key;

internally decrypting said encrypted column with said working key;

processing said query; and

returning an answer to said query to the user.

8. (Currently Amended) A ~~program storage device readable by machine,~~ computer readable medium tangibly embodying a program of instructions executable by ~~the machine~~ a computer to perform method steps for internally encrypting data in a relational database, said method steps comprising:

providing a database engine having encryption as a database kernel feature;

providing a security dictionary comprising one or more security catalogs;

receiving data from a user;

associating said data with a database column and at least one authorized user;

generating a working encryption key;

internally encrypting said working encryption key within said database engine using a public key from an authorized user;

storing said encrypted working key in a security catalog; and

internally encrypting said data within said database engine using said working key.

9. (Currently Amended) The ~~program storage device~~ computer readable medium of claim 8 further comprising the step of generating a private key needed to decrypt said encrypted working key.

10. (Currently Amended) The ~~program storage device~~ computer readable medium of claim 9 wherein said public key is a password and is used by the system to look up said private key.

11. (Currently Amended) The ~~program storage device~~ computer readable medium of claim 8 wherein said step of associating said data with a database column and a user is accomplished with an extended SQL syntax and further comprises the step of creating a relational database object comprising:

the identity of said authorized users; a relational database table;

the identity of said column within said relational database table; and

one or more security flags, said flags indicating user privileges to access said data.

12. (Currently Amended) The ~~program storage device~~ computer readable medium of claim 8 wherein said working key is provided by a user.

13. (Currently Amended) The ~~program storage device~~ computer readable medium of claim 8 wherein said working key is randomly generated.

14. (Currently Amended) The ~~program storage device~~ computer readable medium of claim 8 further comprising the steps of:

receiving a query that involves an encrypted column and a private key from a user;

checking the ownership of [an] the encrypted column using said security catalog to verify the user is authorized;

internally decrypting said encrypted working encryption key with said private key;

internally decrypting said encrypted column with said working key;

processing said query; and

returning an answer to said query to the user.

15. (Previously Presented) The method of claim 1 further comprising the step of writing the encrypted data into a database disk page, after the step of internally encrypting said data within said database engine using said working key.

16. (Currently Amended) The ~~method~~ computer readable medium of claim 8 further comprising the step of writing the encrypted data into a database disk page, after the step of internally encrypting said data within said database engine using said working key.

17. (Previously Amended) A method of internally creating an index for encrypted data, comprising the steps of:

fetching encrypted data pages from storage;

computing a data encryption/decryption key;

decrypting the data to form plaintext data pages;

using said plaintext data pages, building an index and forming index pages; and

encrypting said index pages.

18. (Previously Presented) A method of extending the core SQL statements to integrate encryption as a core feature into a relational database system, comprising the steps of:

adding ENCRYPTION clause to a CREATE TABLE statement;

adding USER clause to the CREATE TABLE statement;

adding ENCRYPTION clause to an ALTER TABLE statement;

adding KEY clause to an INSERT statement;

adding KEY clause to a SELECT statement;

adding UPDATE clause to a CREATE USER statement; and

modifying core SQL statements to integrate encryption and key management as a core database feature supported internally by query compilation and execution components of a database system.